IN THE CLAIMS:

1. (CURRENTLY AMENDED): A method for fast reboot of a computer having an attached disk array and an internal random access memory (RAM) comprising the steps of:

retaining a copy of an operating system kernel at ~~on~~ a reserved storage location of the RAM;

performing predetermined reboot operations with a boot mechanism; and

reloading ~~the~~ an operative operating system at a location in the RAM based upon the copy of the operating system kernel retained at the reserved storage location, after the step of performing the predetermined reboot operations.

2. (ORIGINAL): The method as set forth in claim 1 wherein the boot mechanism is adapted to perform predetermined full reboot steps based upon a full reboot instruction from the operating system, and wherein the predetermined reboot operations omit the predetermined full reboot steps when the operating system is reloaded from the reserved storage location based upon a warm reboot instruction so as to perform a less-than-full reboot.

3. (CURRENTLY AMENDED): The method as set forth in claim 2 wherein the predetermined full reboot steps include loading into the memory a copy of the operating system kernel from the attached disk array ~~on-disk data from which the operating system kernel is reloaded~~, and the predetermined full reboot steps further include at least one of (a) fully clearing of the RAM, including the reserved storage location, and (b) fully testing the RAM.

4. (ORIGINAL): The method as set forth in claim 3 wherein the predetermined full reboot steps further include testing at least one of an LCD display chip and a Serial Input/Output (SIO) chip.

5. (CURRENTLY AMENDED): The method as set forth in claim 2 wherein the copy of ~~in-memory data~~ the operating system kernel comprises a compressed image of the operating system kernel adapted to be uncompressed and extracted to form the operative operating system kernel at the location in the RAM.

6. (CURRENTLY AMENDED): The method as set forth in claim 5 wherein the step of reloading includes loading the compressed image of the operating system kernel from the reserved storage location to a space within the RAM and thereafter uncompressing and extracting the operative operating system kernel into the RAM from the compressed image of the operating system loaded into the RAM.

7. (ORIGINAL): The method as set forth in claim 6 further comprising, after the step of uncompressing and extracting, freeing-for-overwrite the space within the RAM into which the compressed image of the operating system kernel is loaded.

8. (CURRENTLY AMENDED): The method as set forth in claim 6 wherein one of the predetermined full reboot steps comprises copying the compressed image of the operating system kernel into the reserved storage space from the compressed image of the operating system kenrel in the RAM.

9. (CURRENTLY AMENDED): The method as set forth in claim 2 wherein one of the predetermined reboot steps comprises loading the ~~in-memory copy of the data~~ operating

system kernel at the reserved storage location based upon a ~~disk-stored~~ copy of the operating sytem kernel ~~data~~ on the attached disk array ~~of disks~~.

10. (ORIGINAL): The method as set forth in claim 2 further comprising providing a warm reboot instruction in response to a condition that enables the less-than-full reboot of the computer.

11. (ORIGINAL): The method as set forth in claim 10 wherein the condition includes at least one of a user-generated warm reboot command and a predetermined software panic condition that can be repaired by the less-than-full reboot of the computer.

12. (ORIGINAL): The method as set forth in claim 10 wherein the step of providing the warm reboot instruction includes setting a flag in the boot mechanism from a full reboot state to a warm reboot state.

13. (ORIGINAL): The method as set forth in claim 10 further comprising reverting to the full reboot if the copy of the data is corrupted.

14. (CURRENTLY AMENDED): A system for performing a fast reboot of a computer having an attached disk array ~~of attached disks~~ and an internal memory comprising:

a boot mechanism that carries out full reboot operations on the computer, the boot mechanism including a flag adapted to indicate performance by the boot mechanism of either (a) the full reboot operations or (b) warm reboot operations wherein at least one of the full reboot operations is skipped, including a loading of data from the ~~array of disks~~ disk array that generates the operating system kernel at a predetermined kernel location in the memory.

15. (PREVIOUSLY PRESENTED): The system as set forth in claim 14 wherein the full reboot operations that are skipped include a full test of the memory, a zeroing of the memory, and a shutdown of a processor of the computer .

16. (PREVIOUSLY PRESENTED): The system as set forth in claim 14 wherein the memory includes a reserved storage space that stores a copy of data from which the operating system is loaded into the predetermined location in the memory during the warm reboot operations, the boot mechanism being adapted to retain uncleared the reserved storage space during the warm reboot operations, and being adapted to clear the reserved storage space during the full reboot operations.

17. (ORIGINAL): The system as set forth in claim 16 wherein the copy of the data stored in the reserved storage space is a compressed image of the operating system kernel and the boot mechanism is adapted to load the compressed image of the operating system kernel from the reserved storage space into a space in the memory outside the reserved storage space, and further comprising a boot loader that subsequently uncompresses and extracts the compressed image of the operating system kernel at the space in the memory outside the reserved storage space to generate the operating system kernel at the predetermined kernel location in the memory.

18. (CURRENTLY AMENDED): A computer-readable medium including program instructions executing on a computer for fast reboot of a computer having an attached disk array and an internal random access memory (RAM), the program instructions performing the steps of:

retaining a copy of an operating system kernel at ~~in memory data, from which a operating system kernel is reloaded, on~~ a reserved storage location of the RAM;

performing predetermined reboot operations with a boot mechanism; and

reloading ~~the~~ an operative operating system kernel at a location in the RAM based upon the copy of the operating system kernel retained at the reserved storage location after the step of performing the predetermined reboot operations.

19. (ORIGINAL): The computer-readable medium as set forth in claim 18 wherein the boot mechanism is adapted to perform predetermined full reboot steps based upon a full reboot instruction from the storage system, and wherein the predetermined reboot operations omit the predetermined full reboot steps when the operating system is reloaded from the reserved storage location based upon a warm reboot instruction so as to perform a less-than-full reboot.

20. (CURRENTLY AMENDED): The computer-readable medium as set forth in claim 19 wherein the predetermined full reboot steps include loading into the memory a copy of ~~on-disk data from which~~ the operating system kernel ~~is reloaded~~ from the attached disk array, and the predetermined full reboot steps further include at least one of (a) fully clearing of the RAM, including the reserved storage location, and (b) fully testing the RAM.

21. (ORIGINAL): The computer-readable medium as set forth in claim 20 wherein the predetermined full reboot steps further include testing at least one of an LCD display chip and a Serial Input/Output (SIO) chip.

22. (CURRENTLY AMENDED): A method for carrying out a warm reboot on a computer having a random access memory (RAM) and an attached disk array ~~of attached disks~~ that skips predetermined full reboot steps comprising the steps of:

directing a boot mechanism that carries out the full reboot steps to undergo a warm reboot;

retaining an uncleared reserved storage space in the RAM for storing a copy of an ~~so as to store in-memory data from which the~~ operating system kernel ~~is generated~~; and

after predetermined warm reboot steps are performed, generating an operative ~~the~~ operating system kernel in a portion of the RAM outside the reserved storage space from the copy of the operating system kernel in ~~memory data in a portion of the RAM outside~~ the reserved storage space.

23. (CURENTLY AMENDED): The method as set forth in claim 22 wherein the step of generating ~~the operating system kernel~~ includes uncompressing a compressed image of the operating system kernel.

24. (CURENTLY AMENDED): The method as set forth in claim 23 wherein the step of generating further includes copying the ~~in-memory data to form the~~ compressed image of the operating system kernel ~~at~~ to a location within the portion of the RAM outside the reserved storage location.

25. (CURENTLY AMENDED): The method as set forth in claim 23 wherein the step of generating includes performing an error check on the ~~data~~ copy of the operating system kernel and, if the ~~in-memory data~~ copy of the operating system kernel is corrupted, reverting to the full reboot steps.

26. (CURENTLY AMENDED): The method as set forth in claim 23 wherein the full reboot steps include (a) clearing the reserved storage space, (b) loading, from the disk array, a copy of ~~on-disk data from which~~ the operating system kernel ~~is generated and~~ (c)

copying the ~~on-disk data~~ copy of the operating system kernel to the reserved storage space. ~~to thereby store the in-memory data~~.

27. (ORIGINAL): The method as set forth in claim 26 wherein the full reboot steps further comprise at least one of (a) fully clearing the memory, and (b) performing a full test of the memory.

28. (ORIGINAL): A method for fast reboot of a computer having the random access memory comprising the steps of:

performing predetermined reboot steps with a boot mechanism; and

reloading an operating system kernel extracted from a stored operating system kernel image, the operating system kernel image being stored in a reserved location of the random access memory prior to the step of performing the predetermined reboot steps, the reserved location remaining uncleared after the step of performing the predetermined reboot steps.

29. (ORIGINAL): A method for rebooting a computer comprising the steps of:

in response to a predetermined reboot command, selectively reloading an operating system kernel to a memory of the computer from either a first compressed kernel image of the operating system or a second compressed kernel image of the operating system, each of the first compressed kernel image and the second compressed kernel image being located on different non-removable storage media, wherein an access speed for the first compressed kernel image is quicker than an access speed for the second compressed kernel image.

30. (CURRENTLY AMENDED): The method as set forth in claim 29 further comprising storing the first compressed kernel image ~~is~~ on a reserved space of the memory that remains uncleared during a warm reboot process having predetermined warm reboot steps and storing the second compressed kernel on a disk operatively interconnected to the computer.

31. (ORIGINAL): The method as set forth in claim 30 wherein the step of storing the first compressed kernel image includes loading a copy of the second compressed kernel image onto the reserved space during a full reboot process that includes both full reboot steps and the predetermined warm reboot steps.

32. (ORIGINAL): The method as set forth in claim 31 wherein the full reboot steps include clearing the memory, including the reserved storage space and testing each of an LCD display chip and a serial input/output (SIO) chip.

33. (CURRENTLY AMENDED): An operating system for a computer having an operating system kernel stored in a random access memory of the computer comprising:

a warm reboot instruction, responsive to a predetermined reboot condition, that sets an indicator in a boot mechanism of the computer to perform a warm reboot process that includes predetermined boot steps that are fewer than the boot steps performed by the boot mechanism in response to a full reboot instruction; and

a compressed kernel image located at a reserved storage space in the random access memory, the compressed kernel image being adapted to be accessed to reload the compressed kernel image into the random access memory during a warm reboot process.

34. (CURRENTLY AMENDED): The operating system as set forth in claim 33 wherein the boot steps of the full reboot process include mechanisms for fully clearing the random access memory and reloading a compressed kernel image of the operating system from a disk into the random access memory.

35. (CURRENTLY AMENDED): The operating system as set forth in claim 34 wherein the compressed kernel image is located at a reserved storage space in the random access memory that is remote from the area controlled by the operating system and that remains uncleared during the warm reboot process.

36. (ORIGINAL): The operating system as set forth in claim 33 wherein the warm re-boot instruction is based upon a condition in the computer that requires only the boot steps of the warm reboot process and that allows the boot steps of the full reboot process to be skipped.

37. (ORIGINAL): The operating system as set forth in claim 33 wherein the indicator includes a plurality of warm reboot levels that enable each of the boot steps of the full reboot process to be selectively performed or skipped.

38. (PREVIOUSLY PRESENTED) The method as in claim 1, wherein the attached disk array is one or more disks.

39. (CURRENTLY AMENDED) The system as in claim 14, wherein the ~~array of~~ attached disk array ~~disks~~ is one or more disks.

40. (PREVIOUSLY PRESENTED) The computer-readable media as in claim 18, wherein the attached disk array is one or more disks.

41. (CURRENTLY AMENDED) The method as in claim 22, wherein the ~~array of~~ attached disk array ~~disks~~ is one or more disks.

42. (PREVIOUSLY PRESENTED) A method for fast reboot of a computer, the computer having an operating system loaded from a first memory, the method comprising the steps of:

saving a copy of an operating system in a second memory, the second memory having faster access than the first memory;

performing predetermined reboot operations; and

reloading the operating system from the operating system copy in the second memory.

43. (PREVIOUSLY PRESENTED) The method as in claim 42, further comprising the step of: compressing the copy of the operating system in the second memory.

44. (PREVIOUSLY PRESENTED) The method as in claim 43, further comprising the step of: uncompressing the copy of the operating system.

45. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the copy of the operating system is located at a reserved storage space in the second memory that is remote from an area controlled by the operating system and that remains uncleared during the fast reboot process.

46. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the step of performing predetermined reboot operations further comprises skipping at least one step selected from the group consisting of: performing a full memory test, reloading an operating system from the first memory, clearing system memory, testing an LCD display chip, and testing a serial input/output (SIO) chip.

47. (PREVIOUSLY PRESENTED) The method as in claim 42, further comprising: setting a flag indicating the fast reboot.

48. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the second memory is a volatile memory.

49. (PREVIOUSLY PRESENTED) The method as in claim 48, wherein the second memory is a RAM.

50. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the second memory is a non-volatile memory.

51. (PREVIOUSLY PRESENTED) The method as in claim 50, wherein the second memory is an NVRAM.

52. (PREVIOUSLY PRESENTED) The method as in claim 50, wherein the second memory is a flash memory.

53. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the first memory is a disk memory.

54. (PREVIOUSLY PRESENTED) A computer for performing a fast reboot, the computer comprising:

a first memory;

an operating system loaded from the first memory;

a second memory to save a copy of the operating system, the second memory having faster access than the first memory; and

a boot mechanism to perform predetermined reboot operations, including reloading the operating system from the operating system copy in the second memory.

55. (PREVIOUSLY PRESENTED) The computer as in claim 54, wherein the operating system is compressed in the second memory.

56. (PREVIOUSLY PRESENTED) The computer as in claim 55, wherein the operating system is uncompressed by the boot mechanism.

57. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a reserved storage space in the second memory that is remote from an area controlled by the operating system and that remains uncleared during the fast reboot process, the reserved storage space to save the copy of the operating system.

58. (PREVIOUSLY PRESENTED) The computer as in claim 54, wherein the predetermined reboot operations further comprise skipping at least one step selected from the

group consisting of: performing a full memory test, reloading an operating system from the first memory, clearing system memory, testing an LCD display chip, and testing a serial input/output (SIO) chip.

59. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a flag to indicate the fast reboot.

60. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a volatile memory as the second memory.

61. (PREVIOUSLY PRESENTED) The computer as in claim 60, further comprising: a RAM as the second memory.

62. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a non-volatile memory as the second memory.

63. (PREVIOUSLY PRESENTED) The computer as in claim 62, further comprising: an NVRAM as the second memory.

64. (PREVIOUSLY PRESENTED) The computer as in claim 62, further comprising: a flash memory as the second memory.

65. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a disk memory as the first memory.

66. (PREVIOUSLY PRESENTED) A computer for performing a fast reboot, the computer comprising:

a first memory;

an operating system loaded from the first memory;

means for saving a copy of an operating system in a second memory, the second memory having faster access than the first memory;

means for performing predetermined reboot operations; and

means for reloading the operating system from the operating system copy in the second memory.

67. (PREVIOUSLY PRESENTED) The computer as in claim 66, further comprising: a volatile memory as the second memory.

68. (PREVIOUSLY PRESENTED) The computer as in claim 66, further comprising: a non-volatile memory as the second memory.

69. (PREVIOUSLY PRESENTED) The computer as in claim 66, further comprising: a disk memory as the first memory.

70. (PREVIOUSLY PRESENTED) A computer readable media, comprising: said computer readable media containing instructions for execution in a processor on a computer having an operating system loaded from a first memory, the instructions for the practice of the method of,

saving a copy of an operating system in a second memory, the second memory having faster access than the first memory;

performing predetermined reboot operations; and

reloading the operating system from the operating system copy in the second memory.

71. (PREVIOUSLY PRESENTED) Electromagnetic signals propagating on a computer network, comprising: said electromagnetic signals carrying instructions for execution in a processor on a computer having an operating system loaded from a first memory, the instructions for the practice of the method of,

saving a copy of an operating system in a second memory, the second memory having faster access than the first memory;

performing predetermined reboot operations; and

reloading the operating system from the operating system copy in the second memory.